REMARKS

I. Summary of the Office Action.

Claims 1-5 are pending in the above-referenced case.

The Examiner has made objections to the drawings and the specification. The Examiner has provisionally rejected claims 1-5 on grounds of nonstatutory double patenting. The Examiner has rejected claims 1-5 under 35 U.S.C. § 101 as being directed to non-statutory subject matter, and under 35 U.S.C. § 102(b) as being anticipated by Viega et al., "ITS4: A Static Vulnerabilities Scanner for C and C++ Code," 2000 (hereinafter, "Viega").

II. Summary of Applicants' Reply.

Applicants has amended the drawings and the specification to overcome the Examiner's objection. The amendments to drawings and specification are similar to amendments made and accepted for U.S. patent application "Method and System for Detecting Vulnerabilities in Source Code," Serial No. 10/825,007, which faced a similar objection.

Claims 1-5 are cancelled without prejudice. New claims 6-11 are added. The new claims are supported by, for example, paragraphs [0013] – [0105] and [0115] – [0122] of the specification.

Terminal disclaimers are attached herewith to overcome the rejection of claims on grounds of nonstatutory double patenting.

Reconsideration of this application is respectfully requested.

III. The Objections to the specification and the drawings.

Applicants have amended the drawings and the specification to address the Examiner's objections and rejection. Applicants have amended paragraph [0112] to include the following new reference numbers for Figure 12: reference numbers 150, 152, and 154 refer to the rectangles; reference numbers 156, 158, 160, and 162 refer to the ovals; and reference numbers 164, 166, 168, 170, 172, 174, 176 and 176 refer to the arrows. The Replacement Sheet for Figure 12 showing those reference signs is attached hereto. Replacement Sheet for Figure 13

shows the reference signs 134, 136, 138, 140, and 142. Applicants have also amended paragraph [0001] to include the missing serial numbers and to amend the specification to incorporate by reference the contents of provisional patent application Serial No. 60/464,019, to which the present application claims priority under 35 U.S.C. §119(e).

IV. The Rejection of Claims on the Ground of Nonstatutory Double Patenting.

Claims 1-5 were provisionally rejected under obviousness-type double patenting of co-pending applications 10/825,007 and 10/824,865. Terminal disclaimers are filed in conjunction with this paper to obviate any provisional rejection of the new claims.

V. The Rejection of Claims under 35 U.S.C. § 101.

Claims 1-5 were rejected under 35 U.S.C. § 101. The new claims recite relevant language that is similar to that used in related U.S. patent application "Method and System for Detecting Vulnerabilities in Source Code," Serial No. 10/825,007, which faced a similar rejection. That case has since been determined to be recited in an appropriate manner. Consequently, Applicants believe the new claims are likewise directed to proper subject matter under 35 U.S.C. § 101.

VI. The Rejection of Claims under 35 U.S.C. § 102(b).

Claims 1-5 were rejected under 35 U.S.C. §102(b) as being anticipated by Viega. Viega discloses a tool, called ITS4, that performs static analysis on source code for finding vulnerabilities in source code such as buffer overflows and race conditions. In Viega, vulnerability detection is achieved by a lexical analysis of the source code. For example, as disclosed in Viega, "ITS4 breaks a non-preprocessed file into a series of lexical tokens, and then matches patterns in that stream of tokens." Viega, Section 3.1, 1st paragraph. Viega also discloses that other types of static analysis methods, *e.g.*, methods that involves source code parsing, are not used in the ITS4 tool. *See, e.g.*, Viega, Section 3.1.1, 1st paragraph, which states that "we chose not to use a 'real parser.'" In Section 3.1.2 of Viega, it is further disclosed that the ITS4 tool is not designed to be used on complete, semantically valid programs, but on

incomplete code segments, so that the tool can support interactive programming environments. This shows that the ITS4 tool is limited to lexical analysis.

In contrast, new claim 6 recites:

- "executing computer instructions to analyze the source code listing to semantically analyze arguments of the identified routine calls to determine routine calls that possess privilege escalation vulnerabilities using the pre-specified ranges of values."

A semantic analysis of arguments of routine calls as required in new claim 6 is beyond the scope of Viega. As discussed above, the ITS4 tool disclosed in Viega performs a lexical analysis, and is limited to lexical analysis. Semantic analysis of arguments is nowhere to be found in Viega.

Also, new claim 6 is directed to a method of "detecting privilege escalation vulnerabilities in a pre-existing source code listing ... wherein a privilege escalation vulnerability is an uncontrolled escalation of system privileges that allows unauthorized access to system resources." As explained in Paragraph [0115] of the specification, "[p]rivilege escalation vulnerabilities can arise when an application with a high level of system privileges can be made to perform actions outside of the intended design, allowing an outside party to gain privileged access to the system that they would not otherwise posses." Privilege escalation is a type of vulnerability that is different from buffer overflows and race conditions. Viega, however, only discloses the detection of buffer overflow vulnerabilities and race condition vulnerabilities in source code. In rejecting original claim 2, the Examiner stated that column 4 of section 1 of Viega discloses the detection of privilege escalations. However, column 4 of section 1 of Viega only discusses a vulnerability known as file-based race conditions. Nowhere in Viega are privilege escalation vulnerabilities discussed. Hence, Viega does not disclose the technique of detecting privilege escalation vulnerabilities as required by claim 6.

In addition, claim 6 also recites:

- "executing computer instructions to provide pre-specified ranges of values for arguments of routines in the list that cause privilege escalation vulnerabilities"

Viega does not disclose this feature either. Viega does not teach or suggest the use of pre-specified ranges of values for arguments of routines for vulnerability detection.

For the above reasons, Viega does not anticipate claim 6. By at least the same reason, claims 7-11 are patentable in view of Viega. Further, dependent claims 7-8 and 10-11 include additional features not disclosed by Viega. For example, claim 7 recites "analyzing the source code listing to create computer models of the arguments, each model specifying a range of values that each corresponding argument can take when the source code listing is executed," and claim 8 further recites the creation of operand models "specifying a range of values of each corresponding operand as a result of operand transformations expressed in the source code listing." These additional features are not taught or suggested by Viega.

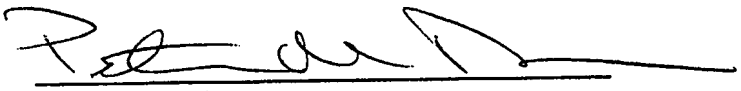
Therefore, applicants respectfully request that new claims 6-11 be allowed.

VII. Conclusion.

For the reasons stated above, we believe that the amended claims are allowable.

The Commissioner is hereby authorized to charge any required fees to our Deposit Account No. 08-0219. Please apply any charges not covered, or any credits, to Deposit Account No. 08-0219.

Respectfully submitted,



Date: October 18, 2007

Peter M. Dichiara
Reg. No. 38,005

Wilmer Cutler Pickering Hale and Dorr LLP
60 State Street
Boston, MA 02109
Telephone: (617) 526-6466
Facsimile: (617) 526-5000